This is a pre publication draft of paper 13-3032 in the proceedings of the Transportation

Research Board 92nd Annual Meeting, January 13-17, 2013, Washington, D.C.

Hidden Markov Models for Vehicle Tracking with Bluetooth

John D. Lees-Miller
Transportation Research Group
University of Southampton
Southampton, SO17 1BJ, United Kingdom
J.Lees-Miller@soton.ac.uk
(corresponding author)

R. Eddie Wilson
Transportation Research Group
University of Southampton
Southampton, SO17 1BJ, United Kingdom
R.E.Wilson@soton.ac.uk

Simon Box
Transportation Research Group
University of Southampton
Southampton, SO17 1BJ, United Kingdom
S.Box@soton.ac.uk

Word Count: 3702 + 1250 (5 figures) = 4952

March 25, 2014

Abstract

Bluetooth is a short range communication protocol. Bluetooth-enabled devices can be detected using road-side equipment, and each detected device reports a unique identifier. These unique identifiers can be used to track vehicles through road networks over time. The focus of this paper is on reconstructing the paths of vehicles through a road network using Bluetooth detection data. A method is proposed that uses Hidden Markov Models, which are a well-known tool for statistical pattern recognition. The proposed method is evaluated on a mixture of real and synthetic Bluetooth data with GPS ground truth, and it outperforms a simple deterministic strategy by a large margin (30%–50%) in this case.

1. INTRODUCTION

Bluetooth is a protocol for short range wireless communication between electronic devices, such as mobile phones, computers and in-car electronics (*1*). Bluetooth devices that are in 'discoverable' mode can be detected from distances of roughly 100m, and when a device is detected, it reports an identifier that uniquely identifies it. Bluetooth detectors can be built using inexpensive off-the-shelf components and deployed at road-side to collect these unique identifiers from devices in passing vehicles.

Bluetooth detection is mainly of interest in vehicle tracking as an alternative (or supplement) to more expensive vehicle detection technologies, such as automatic number plate recognition (ANPR). A pair of Bluetooth detectors can be used to accurately estimate the travel time between the detectors as the time elapsed between the detection of the same device (according to its unique identifier) at one detector and then the other (*2*, *3*). Penetration of discoverable Bluetooth devices in road vehicles varies widely, but it is presently on the order of 10% (*2*), which has proved to be enough to infer accurate travel times. If several detectors are deployed throughout a road network, they can be used along with other road-side sensors, such as inductive loops and ANPR, for traffic assignment and to infer origin-destination matrices (*4*). The same technology is also widely used to track pedestrians (*5*).

The main challenges in using Bluetooth for vehicle tracking are:

1. The position of a detected vehicle is not known precisely. The time of detection is known precisely, but the device can be anywhere within the detection radius of the detector at this time. This radius can be reduced (to increase precision) by tuning antenna characteristics and transmission power levels, but this leads to the next challenge.

2. A device may pass by a detector without being detected. This is due mainly to random delays in the detection process, which can range up to 10s even under ideal radio conditions (*6*); these delays will be discussed in more detail in section 2. Particularly when a vehicle is moving quickly, it can easily pass through the detection radius without being detected. For example, at 22m/s (80km/h; 50mph), a 100m detection radius allows only 5s for detection.

The problem addressed in this paper is to reconstruct the path of a vehicle through a road network using only Bluetooth detection data. In general, the vehicle's path cannot be recovered with certainty, because of the challenges detailed above, but the most likely path can be computed. The approach taken here is to phrase the problem in the language of Hidden Markov Models (HMMs), which are a well-known and widely used formalism for statistical pattern recognition problems (*7*). The resulting paths may be useful for inferring origin-destination matrices and input for traffic assignment.

Section 2 describes how the problem can be constructed as an HMM and then solved using standard techniques. In section 3, the proposed method is evaluated using data collected on a test track.

2. METHOD

We will begin by introducing the concept of a Hidden Markov Model (HMM), and then we will describe its application to the problem at hand. In an HMM, time is discrete. At each time step, the model is in one of a fixed number of *states*, but we cannot directly observe which one (that is, the state is hidden). Instead, the model emits a *symbol*, which can be observed. For each state, there is an *emission* probability distribution over which symbol the model will emit, and there is a *transition* probability distribution over which state the model will be in for the next time step. The usual setting is that we observe a sequence of symbols emitted by the model over time, and we wish to infer the sequence of states that was most likely to generate that sequence of symbols.

In the case of vehicle tracking, the states are chosen points in the road network. Vehicles move between states (that is, along roads) according to transition probabilities that reflect the structure of the road network and the traffic conditions. Each vehicle (or, more precisely, discoverable Bluetooth device) on the network is considered separately, so in each time step, the symbol that we observe is the name of the detector that detected the vehicle, or 'NONE', if the vehicle was not detected in the current time step. The emission probabilities for each state determine the likelihood that a vehicle will be detected by each detector if it is there for one time step; these probabilities will depend mainly on the state's proximity to each of the detectors, but it may also reflect other site-specific factors, such as line-of-sight.

The transition and emission probabilities are to be learned from the Bluetooth data. This is done using the standard Baum-Welch (BW) algorithm (*7*) for HMMs. This algorithm requires an initial (prior) estimate of the transition and emission probabilities, which it iteratively refines based on the observed data. The data consists of one sequence of symbols for each vehicle over a given interval. Technically, it is assumed that the parameters of the model are stationary over this interval, and that the sequences are independent.

To define the states and the initial transition probabilities, we proceed as follows. The required input is a directed graph G, with nodes V and edges E, that represents the road network and determines the allowed routes. A possible road graph is illustrated in figure 1. Note that a two-way road has one set of nodes and edges for each direction.

The states in the HMM are exactly the nodes in the road graph, and the transition probabilities will be constrained so that in each time step a vehicle can only transition to a nearby state in the road graph. Let τ be the length of one time step, in seconds, and let t_{uv} be the shortest time required to travel from state u to state v, also in seconds. The t_{uv} can be obtained by computing shortest paths through the road network and making an assumption on the vehicles' maximum speed (possibly based on posted speed limits). The states reachable from state u are then the states with $t_{uv} \leq \tau$.

There are three types of states: traffic can enter the graph at *source* states, traverse one or more *interior* states, and then exit at *sink* states. Let S, N and T be sets of source, interior and sink states, respectively, so $V = S \cup N \cup T$. When a vehicle reaches a sink state, it assumed that it is undetectable (out of range or turned off), and it remains in the sink state until it re-enters the graph at some source state. The transition probabilities for a sink state will typically exhibit a large probability of remaining in the sink state, and smaller probabilities of returning to various source states. To parameterise this, define for each sink u a positive *weight* w_u that contributes to the probability of remaining in the sink state, and for each source state v, a non-negative weight w_{uv} that contributes to the probability of re-entering the network at v. These weights can be taken to be uninformative (for example, by setting $w_{uv} = 1$ for all u and v and setting w_u to a large number), or they can be set to reflect historical trends or site-specific knowledge (for example, if a sink state leads to a

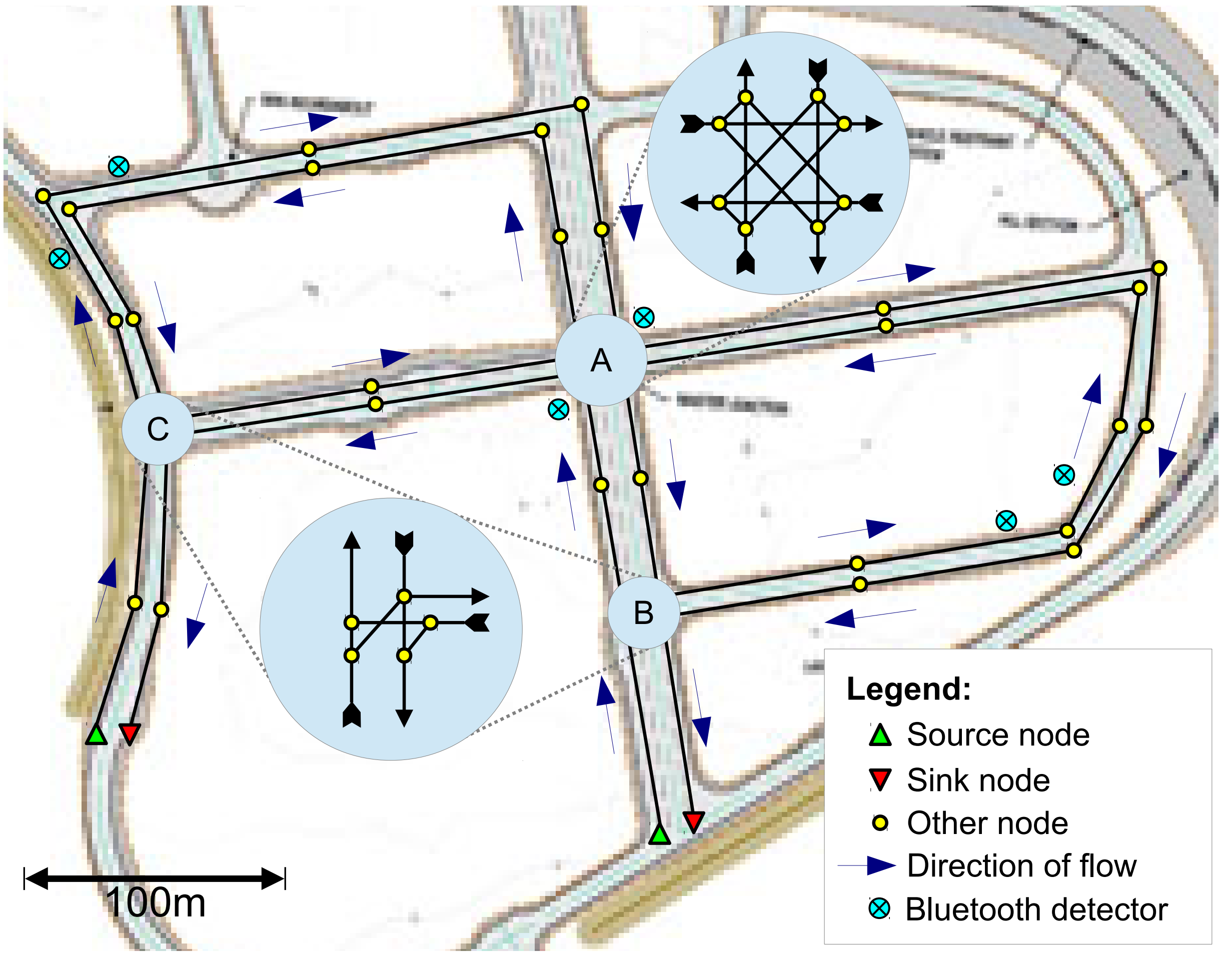

FIGURE 1 Example of a road graph for the 'InnovITS ADVANCE' test track used in section 3. Junction A is signalised; junctions B and C are not. Vehicles were restricted to the figure-of-eight during the trials; the roads in and out were coned off. The nodes and edges at junctions are such that U-turns are not allowed, but all other turns are allowed.

multi-storey car park, it is very likely that vehicles will re-enter from one of that car park's source states).

Putting the road graph constraints and the sink weights together, the relative likelihood of a transition from any state u to any state v is given by

$$\bar{a}_{uv} = \begin{cases} 1, & u \in S \cup N, v \in N \cup T, t_{uv} \leq \tau \quad (1) \\ w_u, & u \in T, v \in T, u = v \quad (2) \\ w_{uv}, & u \in T, v \in S \quad (3) \\ 0, & \text{otherwise} , \quad (4) \end{cases}$$

and the initial transition probabilities a_{uv} can then be obtained by normalising these for each state, that is

$$a_{uv} = \bar{a}_{uv} / \sum_{v} \bar{a}_{uv}.$$

Case (1) allows vehicles to move only to nearby states; if states u and v are too far apart ($t_{uv} > \tau$), case (4) sets the probability of that transition to zero. Note that when an initial probability is set to zero, the HMM learning algorithm cannot make it non-zero, even if that would be a better fit to the data.

This completes the definition of the states and the transition probabilities; it remains to define the emission symbols and probabilities. Here it is important to recall that vehicles are considered one-at-a-time. Let D be the set of Bluetooth detectors. The set of symbols that the HMM can emit is then $D \cup \{\text{NONE}\}$, where NONE means that the vehicle currently being considered was not detected in the current time step. Here we are assuming that it is unlikely that the same device will be detected by more than one detector in one time step; detectors will usually be far enough apart that this is a reasonable assumption.

The raw data from the Bluetooth detectors for a single vehicle is a sequence of time-detector pairs. These data must be converted to a sequence with one symbol (detector) per time step, as follows. Let i be the index of the current time step, with $i = 0, \ldots, n$ where n is the number of time steps to be considered, and let d_i denote the symbol emitted in time step i. If one or more detectors detected the vehicle in the time interval $[i\tau, (i + 1)\tau)$ then set d_i to be the one that detected it first; otherwise, set $d_i = \text{NONE}$.

The emission probabilities then specify for each state (that is, position in the road network) the probability that a vehicle in that state will be (first) detected by each of the detectors, or by no detector, in a single time step. The relationship between position, dwell time and detection probability is in general complicated and site-specific, but only a simple model is required in order to generate initial estimates of these probabilities; the learning process can than refine the estimates based on the observed data. One such simple model is as follows.

Let s_{ud} be the straight line distance in meters between state u and detector d. It is assumed that the time T_{ud} until a vehicle at node u will be detected by detector d follows an Exponential distribution with rate parameter

$$\lambda_{ud} = \gamma s_{ud}^{-2} \quad (5)$$

where γ is a constant to be chosen. This captures the basic intuition that a detector is more likely to detect a devices that is closer, because the signal strength will be higher. In particular, the inverse square law in (5) is based on the Friis transmission equation.

An important feature that this model does not capture very well is that even at close range (< 10m), there can be a significant detection delay due to channel (radio frequency) hopping. Bluetooth uses channel hopping to mitigate the effects of interference with other Bluetooth devices and also with other devices that use the same frequency band, such as WiFi wireless Internet. The detector works by periodically sending an 'inquiry' message on a pseudo-randomly chosen channel. A device will be detected only if it happens to be listening on that channel at the same time, in which case it receives the inquiry message and transmits its unique identifier (and possibly other information) to the detector. Both the detector and the device cycle through the possible channels (at different rates), and it can take some time before they choose the same one. There are several proposed models of the distribution of delays due to channel hopping under various simplifying assumptions (*6*, *8*). Matters are further complicated by the fact that the inquiry protocol has since changed with the 1.2 version of the Bluetooth standard, making newer devices significantly faster to discover. Our results will show that this model is adequate for our purposes here, but improvements may be possible with a more accurate model.

To apply the above model for a single detector to multiple detectors, we note that the time of the first detection at any of the possible detectors, T_u, is $\min_d T_{ud}$, which is itself an Exponential random variable with mean rate $\lambda_u = \sum_d \lambda_{ud}$. The cumulative distribution function $F_u(t)$ of T_u then gives the probability that a vehicle is detected by some detector within one time step. In particular, $F_u(t) = 1 - \exp(-\lambda_u t)$, and the probability of being detected within one time step is $F_u(\tau)$. The probability that a particular detector d is the first one to detect the vehicle, given that there is at least one detection in the time step, is λ_d/λ_u. The initial estimates of the emission probabilities using this model are then

$$b_{ud} = \begin{cases} (\lambda_d/\lambda_u) F_u(\tau), & d \in D \\ 1 - F_u(\tau), & d = \text{NONE}. \end{cases} \tag{6}$$

The final requirement is to define a distribution over the start state for each vehicle; from its start state, the vehicle's movement is thereafter defined by the transition probabilities. Here we simply take all states as equally likely to be start states.

3. RESULTS

The proposed method is evaluated using data collected at the 'InnovITS ADVANCE' test track near Nuneaton, England on 31 May, 2012. The experiment involved 26 vehicles, all of which were cars except for one van and two motor cycles. Each vehicle was equipped with a 1Hz global positioning system (GPS) data logger (model: QStarz BT-Q1000X) that was also a discoverable Bluetooth device (Bluetooth version 1.2; class 2). GPS traces were recovered for 24 vehicles. The test track was set up as a 'figure-of-eight' with a signalised junction at the center, as shown in figure 1; the other areas of the test track were marked off with traffic cones. The primary purpose of the experiment was to evaluate the performance of several junction control algorithms and a human controller with real drivers in congested conditions, but six Bluetooth detectors (figure 2) were also deployed during the experiment.

The data used here are from two twenty-minute trials. In the first trial, drivers were given prescribed routes to follow, such as to drive around the north west loop counter-clockwise. In the second trial, the drivers were asked only to drive as far as possible in

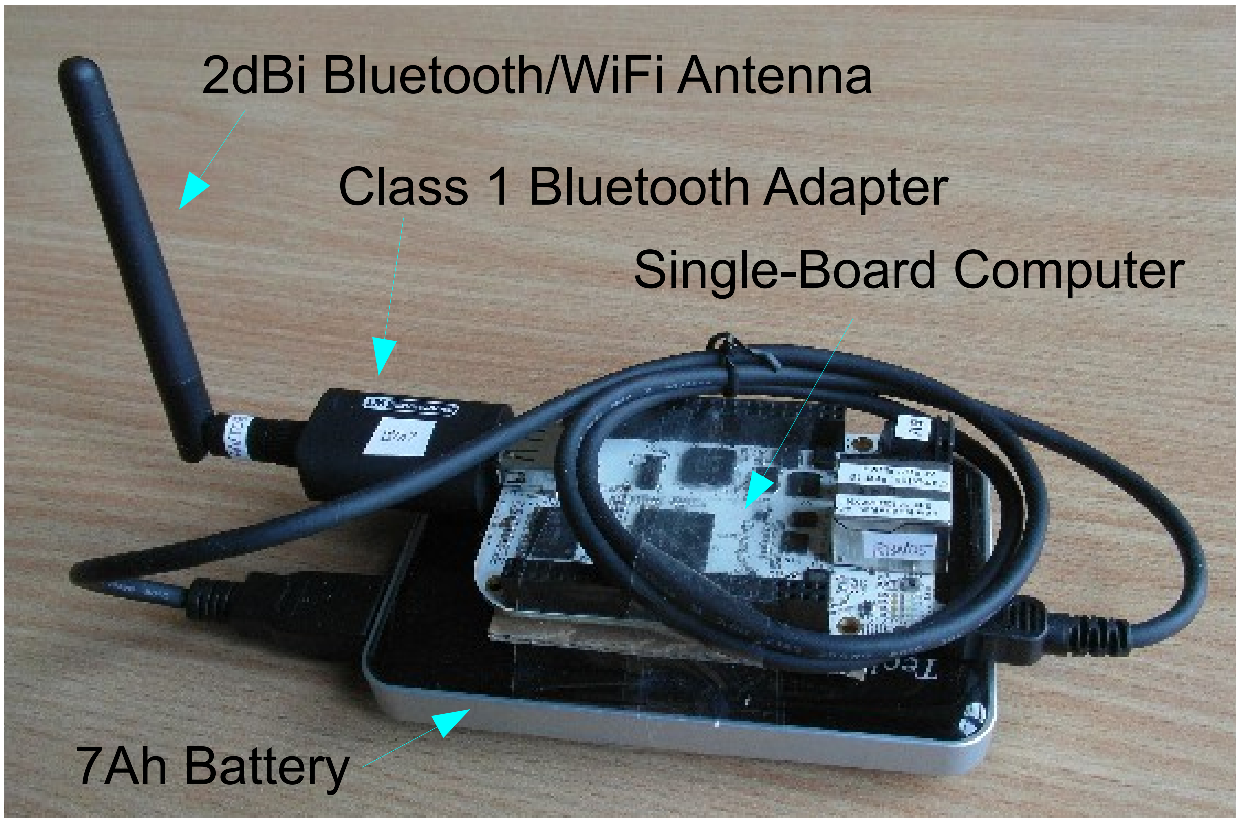

FIGURE 2 A Bluetooth detector used in the trials. The detector is built from off-the-shelf components. The Bluetooth adapter (model: LM Technologies LM540; Bluetooth version 2.1; class 1) is connected to a single-board computer (model: BeagleBone A5), which runs a small program that manages the inquiry process via the BlueZ stack on Linux. The detector runs for one day on battery power. The detectors were mounted in weather-proof enclosures (sandwich boxes) on tripods at roughly 1.5m above the road surface. The antenna is 9cm long.

the time allowed, subject to the site's 30mph speed limit. It is worth remarking that the assumption that the overall traffic pattern is stationary is reasonable within each trial. Data from the first trial was used for preliminary experiments that guided the development of the method and advised on the range of parameters to test. The results presented here are for the data from the second trial, which was used only for evaluation.

There are, however, a number of problems with this dataset, in the context of evaluating the proposed method.

1. The test track is small relative to the usual distances over which Bluetooth detectors are used. The opposing NW-SE corners of the figure-of-eight are only 370m apart. This means that there is more overlap between detection radii than would ordinarily be the case. It also means that the physical separation between states (10m to 30m in these results) may be smaller than would be practical for a larger network.

2. The separation between the pairs of detectors on each end of the figure-of-eight (namely C and D, and E and F in figure 3) was found to be too small to reliably determine in which direction a passing vehicle is driving. In other words, it is difficult to tell from the Bluetooth data alone whether a vehicle is driving one way around the figure-of-eight or the other way around, because the sequences of detections do not look sufficiently different. The proposed HMM method could not resolve this ambiguity, and the result was very large prediction errors that (while they were still below those of the baseline method introduced for comparison below) obscured all other trends. To work around this, we introduce two *simulated* detectors, at points G and H in figure 3. These are deterministic detectors that detect every vehicle within a 20m radius. This makes it much easier to resolve the directional ambiguity, but it may also improve the quality of the results in other ways.

3. The vehicles began each trial already parked on the figure-of-eight, and they remained on the figure-of-eight for the whole trial. This means that the vehicles did not use any source or sink states. For simplicity, the source and sink states (and connecting edges) were removed to leave only the figure-of-eight. The model definition is still as described in section 2, but with $S = T = \emptyset$ in (1–3).

The parameters to be chosen are the separation between states, the time step, the speed limit used to set the initial transition probabilities, and the γ used in the initial emission model (5). Graphs with varying separation between states were generated by constructing a graph with 5m separation between states and then subsampling. The site speed limit was 13.4m/s (30mph), but to allow for occasional overspeed, a speed limit of 20m/s was used. That is, t_{uv} was set to the shortest path distance from u to v, divided by 20m/s. The results presented here use $\gamma = 50$, which gives a detection rate of 99% in 10s at 10m and 5% in 10s at 100m.

Figure 3 compares a vehicle's trajectory as reconstructed from Bluetooth data with the (essentially ground truth) trajectory obtained from its GPS logger. The main purpose of the the HMM is to infer the position of the vehicle between Bluetooth detections. For

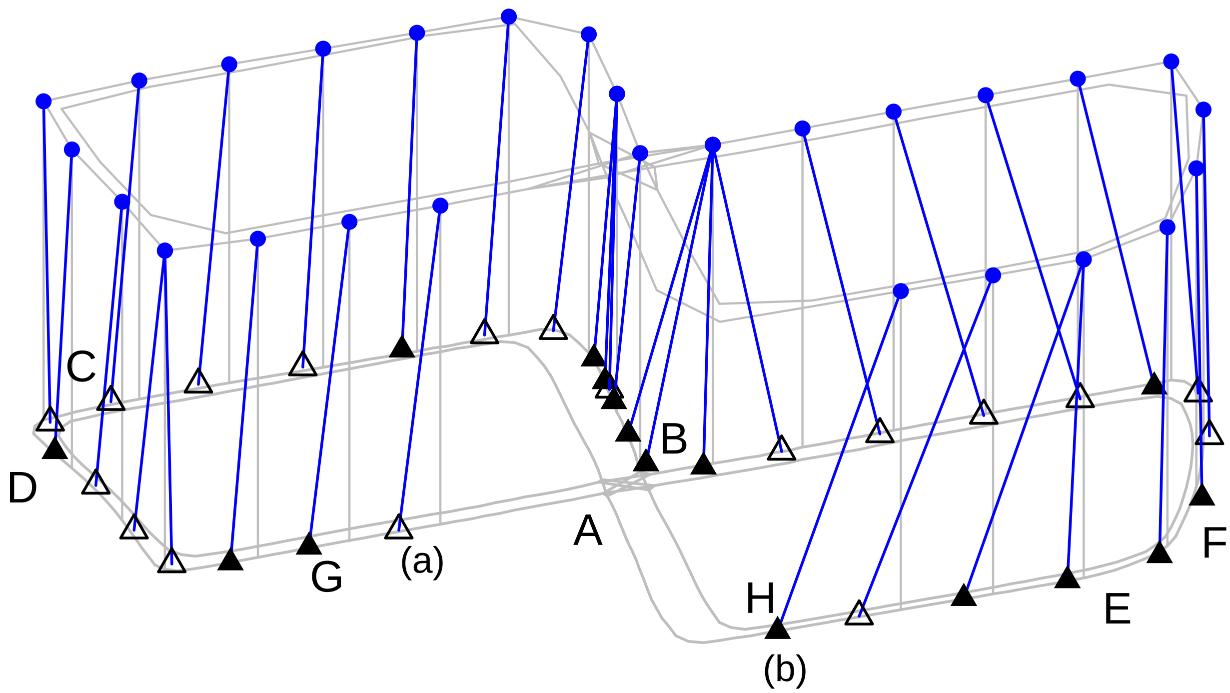

FIGURE 3 Comparison of a vehicle trajectory from GPS with the one reconstructed from Bluetooth data. The vehicle begins at point (a) and drives clockwise (driving on the left) to point (b). The vehicle positions from GPS are shown on the bottom layer (triangles). If the vehicle was detected by a Bluetooth detector, the triangle is filled; otherwise it is empty. The corresponding (as indicated by a connecting line) positions predicted by the HMM are shown on the top layer. Here the state separation is 30m, and $\tau = 3$s.

example, when the vehicle passes between detectors B and F (or possibly it was detected at A or E, since the detection radii overlap significantly) for five time steps (15s), the HMM predicts that it moves one state forward in each time step; its predictions are accurate to within one state, in this example. The HMM used here was the result of 10 Baum-Welch iterations.

Figure 4 shows the progress of the BW algorithm in training an HMM for the second trial for several different parameter settings. The training regime uses 4-fold cross validation (CV): in each fold, three quarters of the 24 vehicles are used for training, the remaining quarter are used for validation. The BW algorithm aims to maximise the likelihood of the training data. It is guaranteed to converge to a local maximum, but it may not find a global maximum; this is why it is important to set the initial transition and emission probabilities carefully. The figure shows that the likelihood of the training data increases with each iteration, but that the likelihood of the validation data reaches a maximum and then decreases. The algorithm should typically be stopped when the likelihood of the validation set begins to decrease; it is likely that further iterations will lead to over fitting. It is important to note that the likelihood is defined with reference to the Bluetooth data only; the GPS data are not used (except indirectly via the simulated Bluetooth detectors, in this case). CV can therefore by used to decide when to stop training, even when ground truth GPS data are not available.

Figure 5 shows the progress of the training in terms of the accuracy of the predicted vehicle positions, measured against the actual GPS positions. It can be seen that the training process reduces the prediction error below that for the initial HMM. Moreover, by comparing with 4, it can be seen that most of this improvement is made by the time the likelihood of the CV validation set begins to decrease (although there is some further decrease in the bottom two panels).

We are not aware of any previous methods for this problem to which we could compare these results. For comparison, a simple deterministic strategy was also evaluated. It is as follows.

1. Let the $c(d)$ denote the closest state to detector d.
2. When the vehicle is detected by detector d_1 at time t_1, predict that it is in state $c(d_1)$ at time t_1.
3. When the vehicle is detected by detector d_2 at time t_2, find the shortest path from $c(d_1)$ to $c(d_2)$ and assume that the vehicle maintains a constant average speed along that path.
4. At the start of the sequence, before we have detected the vehicle the first time, assume it is in $c(d_1)$.
5. At the end of the sequence, assume that the vehicle remains in its last position.

This strategy is the ‘baseline’ line in figure 5, and its performance is consistently worse than the HMM, even before training.

Overall, the best accuracy was achieved with shorter time steps and state separations; experiments with even shorter time steps and separations did not yield much improvement over these results, however, and they did increase the computation time. Computation times for $\tau = 3$ and 10m state separation were on average 67s per iteration for all four CV folds. Training was done using ‘JAHMM’ (version 0.6.2), an open source HMM library.

4. CONCLUSION

This paper described a method for tracking vehicles using data from Bluetooth sensors based on Hidden Markov Models (HMMs). The method was evaluated on a mixture of real and synthetic Bluetooth data. The method was able to reconstruct vehicle trajectories using only Bluetooth data. The proposed approach outperformed a simple deterministic strategy by a large margin (30%–50%) in this case.

There is much scope for future work:

- The model used here was a pure HMM. This meant that certain constraints could not be included. For example, emission probabilities for adjacent states may be tightly coupled, but in this model they are independent parameters. Coupling constraints could considerably reduce the potential for over fitting. Inference would then require a more general Expectation Maximisation algorithm.

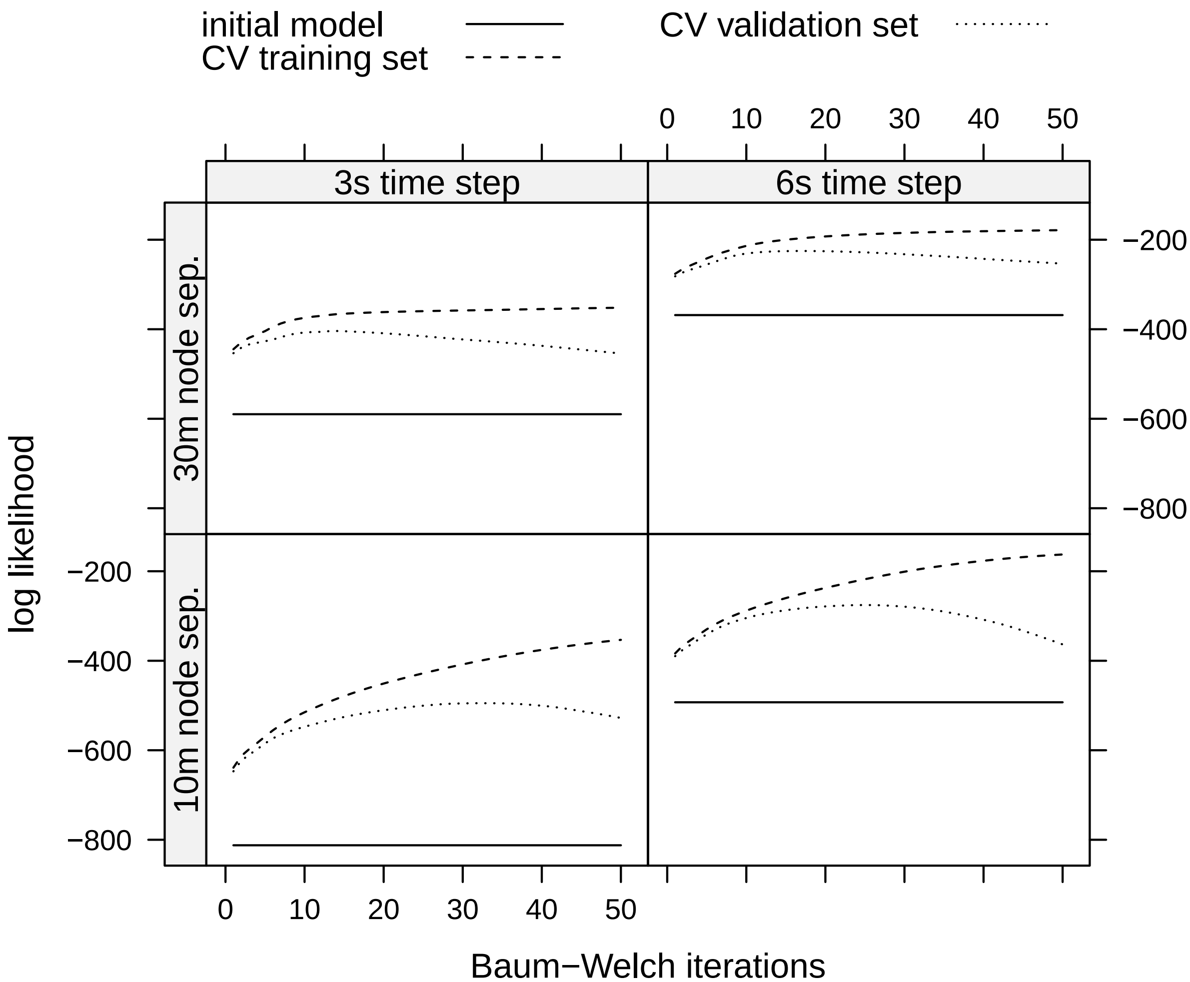

FIGURE 4 Logarithm of the likelihood of the cross-validation training and validation sets over several iterations of the BW algorithm used for training the HMM. While the likelihood of the CV training set always increases, the likelihood of the CV validation set may decrease; this can be used to decide when to stop the training algorithm.

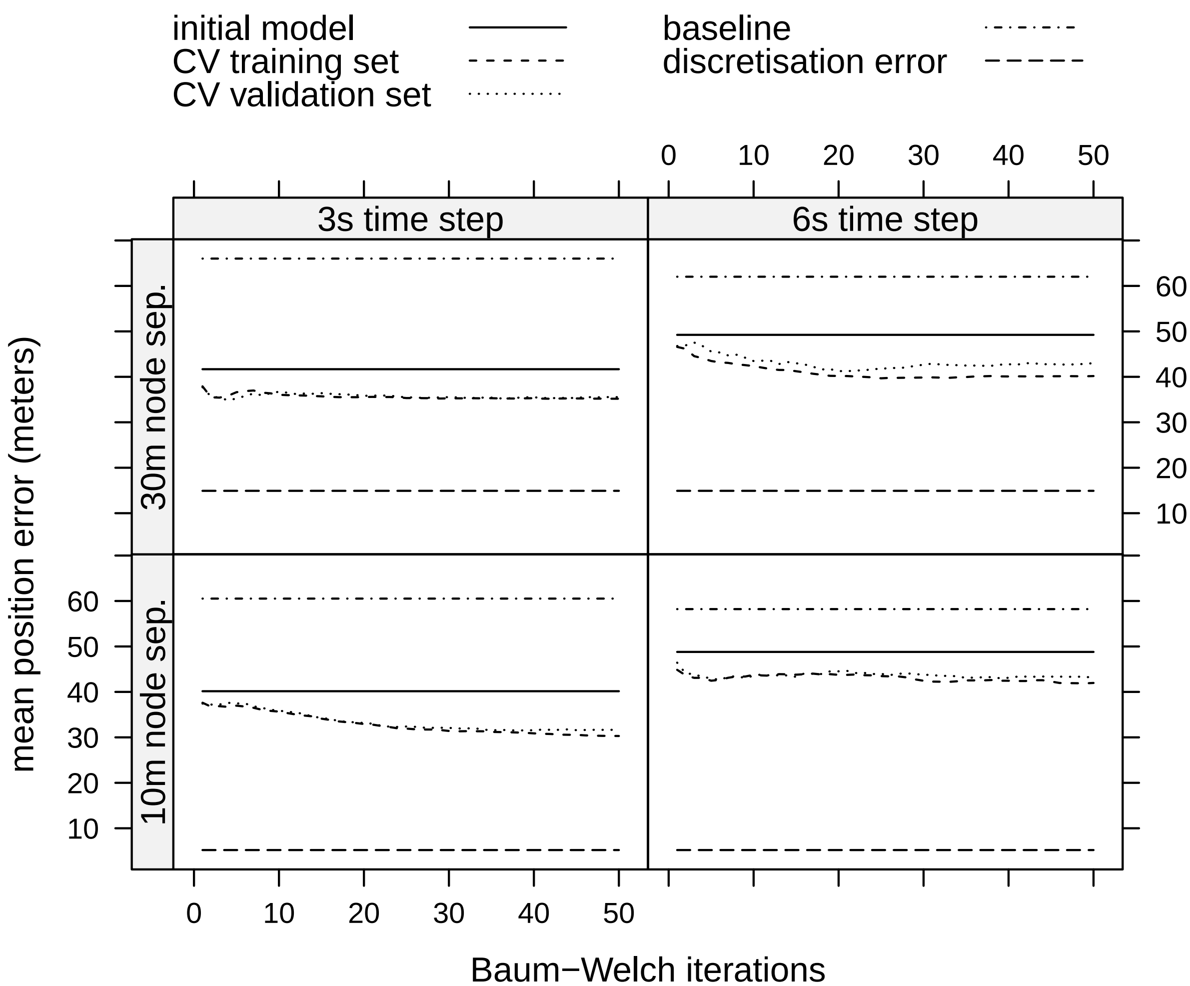

FIGURE 5 Mean prediction error over the course of the training process. The prediction error is the difference between the vehicle position predicted by the HMM and the GPS position. The 'baseline' is a simple deterministic strategy, as described in the text. The 'discretisation error' is half of the nominal state separation; this is a lower bound on the achievable prediction error for a given discretisation of the road network.

- There may be opportunities to share parameters between groups of states, particularly when they are far away from detectors. This would reduce the effective number of parameters in the model.

- The model was evaluated in an 'off line' mode, in which all of the Bluetooth data was available (smoothing). In practice, it may be desirable to run the system in an 'online' fashion (filtering).

- Bluetooth detectors also report a 'received signal strength indicator' (RSSI) for each detection. This is generally correlated with distance, so it may provide extra information about the position of the vehicle when it was detected. One possible way of incorporating this information into this HMM would be to expand the set of symbols to include 'strong' and 'weak' RSSI detection symbols for each detector.

- The method was evaluated on a small road system. It is likely that the method would have to be applied over much larger road networks in practice. The positive results obtained here suggest that evaluation on a larger scale is worthwhile.

REFERENCES

1. Bluetooth Special Interest Group. Specification of the Bluetooth System, version 2.1 + EDR, Jul 2007.

2. Malinovskiy, Y., U.-K. Lee, Y.-J. Wu and Y. Wang. Investigation of Bluetooth-Based Travel Time Estimation Error on a Short Corridor. In *Transportation Research Board 90th Annual Meeting*, 2011.

3. Porter, J. D., D. S. Kim, M. E. Magana, P. Poocharoen and C. A. Gutierrez Arriaga. Antenna Characterization for Bluetooth-based Travel Time Data Collection. In *Transportation Research Board 90th Annual Meeting*, 2011.

4. Barceló, J., L. Montero, M. Bullejos, O. Serch and C. Carmona. A Kalman Filter Approach for the Estimation of Time Dependent OD Matrices Exploiting Bluetooth Traffic Data Collection. In *Transportation Research Board 91st Annual Meeting*, 2012.

5. Malinovskiy, Y., N. Saunier and Y. Wang. Pedestrian Travel Analysis Using Static Bluetooth Sensors. In *Transportation Research Board 91st Annual Meeting*, 2012.

6. Chakraborty, G., K. Naik, D. Chakraborty, N. Shiratori and D. Wei. Analysis of the Bluetooth device discovery protocol. *Wirel. Netw.*, vol. 16, no. 2, pp. 421–436, Feb 2010.

7. Rabiner, L. and B. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986.

8. Salonidis, T., P. Bhagwat, L. Tassiulas and R. LaMaire. Distributed topology construction of Bluetooth personal area networks. In *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213)*. IEEE, 2001, pp. 1577–1586.